EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	83	(nie-xiao$).in.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2007/02/18 23:09
L2	60	(nie-xiaoning$).in.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2007/02/18 23:09
S1	0	((multi?thread$3) near4 (instruction adj1 buffers)) with (vliw or (very adj1 long adj1 instruction adj1 word$1) or (very adj1 long adj1 instruction) or (instruction adj1 (bundle$1 or group$1 or packet$1)))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/22 08:12
S2	0	((multi?thread$3) near4 (instruction adj1 buffers)) same (vliw or (very adj1 long adj1 instruction adj1 word$1) or (very adj1 long adj1 instruction) or (instruction adj1 (bundle$1 or group$1 or packet$1)))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/18 15:25
S3	0	((multi?thread$3) near4 (instruction adj1 buffers)) and (vliw or (very adj1 long adj1 instruction adj1 word$1) or (very adj1 long adj1 instruction) or (instruction adj1 (bundle$1 or group$1 or packet$1)))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/18 15:26
S4	11	((multi?thread$3 or ((multiple or plurality) near4 process$3)) near4 (instruction adj1 buffers)) and (vliw or (very adj1 long adj1 instruction adj1 word$1) or (very adj1 long adj1 instruction) or (instruction adj1 (bundle$1 or group$1 or packet$1)))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/18 15:49
S5	9	(instruction adj1 buffers) near4 (vliw or (very adj1 long adj1 instruction adj1 word$1) or (very adj1 long adj1 instruction) or (instruction adj1 (bundle$1 or group$1 or packet$1)))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/18 15:31
S6	15	(instruction adj1 buffers) with (vliw or (very adj1 long adj1 instruction adj1 word$1) or (very adj1 long adj1 instruction) or (instruction adj1 (bundle$1 or group$1 or packet$1)))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/18 15:33

EAST Search History

S7	20	(instruction adj1 buffers) same (vliw or (very adj1 long adj1 instruction adj1 word$1) or (very adj1 long adj1 instruction) or (instruction adj1 (bundle$1 or group$1 or packet$1)))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/18 15:36
S8	1	("5574939").PN.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/18 15:37
S9	1	("4229790").PN.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/18 15:37
S10	1	("4320453").PN.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/18 15:38
S11	12	((multi?thread$3 or ((multiple or plurality) near4 process$3)) with (instruction adj1 buffers)) and (vliw or (very adj1 long adj1 instruction adj1 word$1) or (very adj1 long adj1 instruction) or (instruction adj1 (bundle$1 or group$1 or packet$1)))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/18 15:51
S12	11	((multi?thread$3 or ((multiple or plurality) near4 process$3)) with (instruction adj1 buffers)) with (superscalar or parallel)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/18 15:52
S13	0	((multi?thread$3 or ((multiple or plurality) near4 process$3)) with (instruction adj1 buffers)) with (superscalar or (parallel adj1 process$1))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/18 15:52
S14	12	((multi?thread$3 or ((multiple or plurality) near4 process$3)) with (instruction adj1 buffers)) with (superscalar or (parallel adj1 process$1) or (instruction adj1 units) or processors)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/18 15:53
S15	0	((multi?thread$3 or ((multiple or plurality) near4 process$3)) with (instruction adj1 buffers)) with (superscalar or (parallel adj1 process$1) or (instruction adj1 units))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/18 15:53

S16	1	("5913925").PN.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/18 15:59
S17	180	(712/206).CCLS.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/22 08:12
S18	0	vliw near4 (instruction adj1 format) near4 length	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/22 08:13
S19	5	vliw with ((instruction adj1 format) near4 length)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/22 08:14
S20	14	vliw near4 (instruction adj1 length)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/22 08:18
S21	5	vliw same ((instruction adj1 format) near4 length)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/22 08:14
S22	18	vliw with (instruction adj1 length)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/22 08:19
S23	0	vliw with (instruction adj1 length adj1 bit$1)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/22 08:20
S24	0	vliw with (instruction near4 (length adj1 bit$1))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/22 08:20
S25	0	vliw with (instruction with (length adj1 bit$1))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/22 08:20
S26	0	vliw same (instruction with (length adj1 bit$1))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/22 08:20
S27	0	vliw same (instruction same (length adj1 bit$1))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/02/22 08:20

S28	187	(712/206).CCLS.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/07/07 14:07
S29	8	(multi?thread$3) near4 (instruction$1 adj1 (buffer$1 or cach$3))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/07/07 14:08
S30	41	parady.in.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/07/07 14:10
S31	194	(712/206).CCLS.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/12/19 12:14
S32	3806	(select$3 or multiplex$3 or choos$3) near4 (buffers near4 (output$1 or data))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/12/19 12:15
S33	3790	(select$3 or multiplex$3) near4 (buffers near4 (output$1 or data))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/12/19 12:15
S34	6	(select$3 or multiplex$3) near4 ((multiple adj1 buffers) near4 (output$1 or data))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/12/19 12:30
S35	15	(multiple adj1 buffers) with ((select$3 or multiplex$3) near4 (output$1 or data))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/12/19 12:39
S36	1	(buffers) near4 ((select$3 or multiplex$3) near4 (output$1 or data)) near4 (processing adj1 units)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/12/19 12:39
S37	0	(buffers) near4 ((select$3 or multiplex$3) near4 (output$1 or data)) near4 (multiple near4 instructions)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/12/19 12:39
S38	3	(buffers) near4 ((select$3 or multiplex$3) near4 (output$1 or data)) near4 (instructions)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/12/19 12:42
S39	0	(parallel near4 instruction$1 near4 fetch$3) near4 (multiplex$3 or select$3) near4 buffers	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/12/19 12:44

S40	0	(parallel near4 instruction$1 near4 fetch$3) with ((multiplex$3 or select$3) near4 buffers)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/12/19 12:44
S41	22	(parallel near4 instruction$1 near4 fetch$3) same ((multiplex$3 or select$3) near4 buffers)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/12/19 12:46
S42	27	(parallel with fetch$3) same ((multiplex$3 or select$3) with buffers)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/12/19 12:46
S43	194	(712/206).CCLS.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/12/19 13:28
S44	1	"6324639".pn.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/12/19 13:46
S45	200	(712/206).CCLS.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2006/05/11 10:30
S46	211	(712/206).CCLS.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2007/01/08 11:52
S47	211	(712/206).CCLS.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2007/02/06 09:06
S48	0	select$3 with (buffer$3 near4 parallel near4 execut$3)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2007/02/06 11:10
S49	0	((thread$3 near4 buffer$1) with (instruction$1 near4 issu$5)) same ((parallel$5 or simulatenous$2 or superscal$5) near4 ((function$5 or execut$5 process$4) near4 (unit$1 or logic$1)))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2007/02/06 11:13
S50	0	((thread$3 near4 buffer$1) with (instruction$1 near4 issu$5)) same ((parallel$5 or simulatenous$2 or superscal$5) with ((function$5 or execut$5 process$4) near4 (unit$1 or logic$1)))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2007/02/06 11:14

S51	50	((thread$3 near4 buffer$1) with (instruction$1 near4 issu$5))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2007/02/06 11:19
S52	69	(thread$3 near4 issu$3) with ((parallel$3 or simultaneous$3) near4 (execut$3 or process$3))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2007/02/06 11:25

THE GUIDE TO COMPUTING LITERATURE

Feedback Report a problem Satisfaction survey

Exploiting choice: instruction fetch and issue on an implementable simultaneous mult processor

Full text   Pdf (1.48 MB)

Authors   Dean M. Tullsen   Dept of Computer Science and Engineering, University of Washington, Box 352350, Seattle, WA
Susan J. Eggers   Dept of Computer Science and Engineering, University of Washington, Box 352350, Seattle, WA
Joel S. Emer   Digital Equipment Corporation, HLO2-3/J3, 77 Reed Road, Hudson, MA
Henry M. Levy   Dept of Computer Science and Engineering, University of Washington, Box 352350, Seattle, WA
Jack L. Lo   Dept of Computer Science and Engineering, University of Washington, Box 352350, Seattle, WA
Rebecca L. Stamm   Digital Equipment Corporation, HLO2-3/J3, 77 Reed Road, Hudson, MA

Additional Information: abstract references citings index terms collaborative colleagues peer to peer

Tools and Actions: Find similar Articles   Review this Article
Save this Article to a Binder   Display Formats: BibTex EndNote ACM Ref

↑ ABSTRACT

Simultaneous multithreading is a technique that permits multiple independent threads to issue multiple instructions each cycle. In previous work we demonstrated the performance potential of simultaneous multithreading, based on a somewhat idealized model. In this paper we show that the throughput gains from simultaneous multithreading can be achieved *without* extensive changes to a conventional wide-issue superscalar, either in hardware structures or sizes. We present an architecture for simultaneous multithreading that achieves three goals: (1) it minimizes the architectural impact on the conventional superscalar design, (2) it has minimal performance impact on a single thread executing alone, and (3) it achieves significant throughput gains when running multiple threads. Our simultaneous multithreading architecture achieves a throughput of 5.4 instructions per cycle, a 2.5-fold improvement over an unmodified superscalar with similar hardware resources. This speedup is enhanced by an advantage of multithreading previously unexploited in other architectures: the ability to favor for fetch and issue those threads most efficiently using the processor each cycle, thereby providing the "best" instructions to the processor.

↑ REFERENCES

Note: OCR errors may be found in this Reference List extracted from the full text article.